

(19)日本国特許庁 (JP)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平8-305664

(43)公開日 平成8年(1996)11月22日

(51) Int.Cl.⁶

識別記号 庁内整理番号
330

F I
G 0 6 F 15/16

技術表示箇所

審査請求 未請求 請求項の数 2 O.L. (全 7 頁)

(21)出願番号 特願平7-105509

(22)出願日 平成7年(1995)4月28日

(71)出願人 000001937

日本電気ホームエレクトロニクス株式会社
大阪府大阪市中央区城見一丁目4番24号

(72) 発明者 赤井 昭則
大阪府大阪
日本電気本
内

(54) 【発明の名称】 C P U監視方法及びC P U監視装置

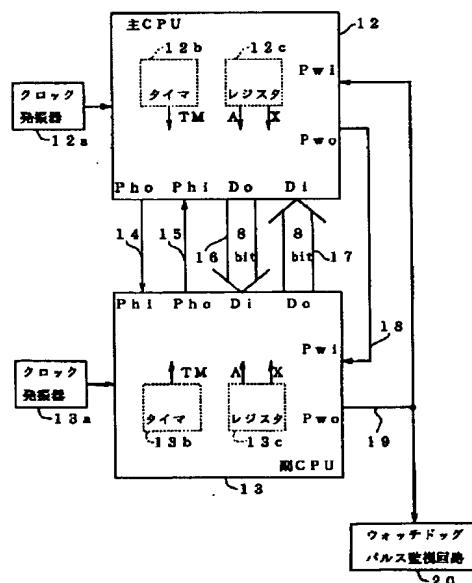
(57) 【要約】

【目的】 それがほぼ同周期の固有のクロック信号に基づいて動作する一対のCPUが、常に互いに共通する同一のループタイムをもってループ処理を実行するよう監視する。

【構成】 それぞれ固有のクロック信号に従って動作する一対のCPU12, 13が、ハンドシェーク線14, 15とウォッチドッグパルス線18, 19を介して一定時間ごとに同期処理を行い、しかも同期処理に続いて主CPU12は自らの同期処理に要した時間(X+Y)をもって、また副CPU13は同期通信線16を介して主CPU12から送られてくる前記時間(X+Y)をもつて、それぞれループタイム目標値TLを補正する。これにより、1ループごとにループタイムを共通の一定時間に規制し、正確に管理されたループタイムのもとCPU12, 13の円滑な処理を保証することができる。

— 本発明のCPU監視装置の一実施例を示すブロック構成図

11 CPU相互監視システム



1

【特許請求の範囲】

【請求項1】 それぞれがほぼ同周期の固有のクロック信号に基づき、それぞれに固有の割り込み処理を織り混ぜて互いにほぼ同様のループ処理プログラムに従って動作する一対のCPUが、それぞれのループ処理プログラム途中の所定ステップにおいてループ処理に要するループタイムが所定のループタイム目標値に一致したことを判定し、前記ループタイムの計時を新たに同時期に再開するよう相互に同期処理を行うとともに、一方のCPUは自らの同期処理に要した時間をもって、また他方のCPUは前記一方のCPUが同期処理に要した前記時間をもって、それぞれ前記ループタイム目標値を補正し、一対のCPUが常に互いに共通する同一のループタイムをもってループ処理を実行するよう監視することを特徴とするCPU監視方法。

【請求項2】 それぞれがほぼ同周期の固有のクロック信号に基づき、それぞれに固有の割り込み処理を織り混ぜて互いにほぼ同様のループ処理プログラムに従って動作し、それぞれのループ処理プログラム途中の所定ステップにおいてループ処理に要するループタイムが所定のループタイム目標値に一致したことを判定してそれぞれウォッチドッグパルスを出力するとともに、前記ループタイムの計時を新たに同時期に再開するようハンドシェーク信号を出して相互に同期処理を行う一対のCPUであって、一方のCPUは自らの同期処理に要した時間をもって、また他方のCPUは前記一方のCPUが同期処理に要した前記時間をもって、それぞれ前記ループタイム目標値を補正し、常に互いに共通する同一のループタイムをもってループ処理を実行するよう監視する前記一対のCPUと、該一対のCPUを相互接続し、前記ウォッチドッグパルス出力を相互に送受する一対のウォッチドッグパルス線と、前記一対のCPUを相互接続し、前記ハンドシェーク信号を送受する一対のハンドシェーク信号線と、前記一対のCPUを接続し、前記一方のCPUが自らの同期処理に要した時間データを前記他方のCPUに送出するデータ通信線とを具備することを特徴とするCPU監視装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、それぞれがほぼ同周期の固有のクロック信号に基づいて動作する一対のCPUが、常に互いに共通する同一のループタイムをもってループ処理を実行するよう監視するCPU監視方法及びCPU監視装置に関する。

【0002】

【従来の技術】 ほぼ同一内容のループ処理を繰り返す主副一対のCPUを監視するため、これら一対のCPUから1回のループ処理を終えるつどウォッチドッグパルスを出力させるとともに、これらのウォッチドッグパルスを外部のウォッチドッグパルス監視回路にて照合させ、

両者のウォッチドッグパルスに周期ずれが発生したときにウォッチドッグパルス監視回路がCPUの暴走を警告するよう構成したCPU監視装置が知られている。この種の従来のCPU監視装置は、仮に一对のCPUを同一のクロック発振器を共通に使用して作動させいわゆる1クロック2CPUのシステム構成を採用した場合、一对のCPU2, 3間でループタイムがずれることはあり得ないが、クロック周波数が仮に本来想定していた例えは16MHzから15MHzに変動したとしても、こうしたクロック発振器自体の誤動作に起因するループタイムの時間ずれについては検出することができなかった。

【0003】 これに対し、例えば図5に示すCPU監視装置1のごとく、主CPU2と副CPU3にそれぞれ専用のクロック発振器2a, 3aを接続し、一方のクロック発振器2a又は3aに生じた動作異常も含め、ループタイムの時間ずれをもってウォッチドッグパルス監視回路4が暴走検知する構成の2クロック2CPU型が提案されている。

【0004】

【発明が解決しようとする課題】 上記従来のCPU監視装置1は、クロック発振器2a, 3aのクロック周期が僅かでも食い違っている場合には、ループ処理を重ねるほどにクロック周期の誤差が堆積してしまうために、それぞれのCPU2, 3は与えられたプログラムを正規に処理してはいても、ループタイムに許容範囲を越える時間ずれが生ずる結果、ウォッチドッグパルス監視回路4の暴走判断とともに主CPU2と副CPU3が動作を停止することがある等の課題があった。そこで、クロック発振器2a, 3a間に存在するクロック周期ずれを考慮し、一步踏み込んだ処置として、ループタイムの時間差が許容範囲にある場合に限り、主CPU2と副CPU3とを1ループごとに適当なタイミングで強制的に同期させる方法も考えられるが、こうした強制同期法を採用した従来の装置は、CPU2, 3それぞれの処理時間に個体差が存在したり、或いは各CPU2, 3の処理途中に挿入される割り込み処理に費やされる時間が異なるといった理由から、暴走以外の原因でウォッチドッグパルス監視回路4が作動してしまいやすく、このため幾重にも安全設計を施した高度のウォッチドッグパルス監視回路が不可欠であり、製造コストの上昇が避けられない等の課題があった。

【0005】

【課題を解決するための手段】 本発明は、上記課題を解決したものであり、それぞれがほぼ同周期の固有のクロック信号に基づき、それぞれに固有の割り込み処理を織り混ぜて互いにほぼ同様のループ処理プログラムに従って動作する一対のCPUが、それぞれのループ処理プログラム途中の所定ステップにおいてループ処理に要するループタイムが所定のループタイム目標値に一致したことを判定し、前記ループタイムの計時を新たに同時期に

再開するよう相互に同期処理を行うとともに、一方のCPUは自らの同期処理に要した時間を持って、また他方のCPUは前記一方のCPUが同期処理に要した前記時間をもって、それぞれ前記ループタイム目標値を補正し、一対のCPUが常に互いに共通する同一のループタイムをもってループ処理を実行するよう監視することを特徴とするものである。

【0007】

【作用】本発明は、それぞれ固有のクロック信号に従って動作する一対のCPUが、一定時間ごとに同期処理を行い、しかも同期処理に統いて一方のCPUは自らの同期処理に要した時間を持って、また他方のCPUは一方のCPUから送られてくる前記時間をもって、それぞれループタイム目標値を補正することにより、1ループごとにループタイムを共通の一定時間に規制し、正確に管理されたループタイムのもと一対のCPUの円滑な処理を保証する。

【0008】

【実施例】以下、本発明の実施例について、図1ないし図4を参照して説明する。図1は、本発明のCPU監視装置の一実施例を示すブロック構成図、図2は、図1に示した回路各部の信号波形図、図3は、図1に示した主CPUの動作を説明するためのフローチャート、図4は、図3に示したウォッチドッグパルス同期処理ステップの具体的な内容を示すフローチャートである。

【0009】図1に示すCPU監視システム11は、説明の便宜上主副の区別を施した一対のCPU12, 13が、互いに専用のクロック発振器12a, 13aが発するほぼ同周期のクロック信号に基づいて動作し、それぞれほぼ同じ動作プログラムに従ってループ処理を実行する。各CPU12, 13は、ループタイム等を計時するタイマ12b, 13bと、両CPU12, 13の同期処理に必要な計数値を保持するレジスタ12c, 13cを内蔵する。また、CPU12, 13は、互いのハンドシェーク信号出力ポートPhoとハンドシェーク信号入力ポートPhiを一対のハンドシェーク信号線14, 15が相互接続し、かつ互いのデータ出力ポートDoとデータ入力ポートDiを8ビットのデータ通信線16, 17が相互接続している。さらに、互いのウォッチドッグパルス出力ポートPwoとウォッチドッグパルス入力ポートPwiがウォッチドッグパルス線18, 19により相互接続されており、副CPU13のウォッチドッグパルス出力ポートWDoと主CPU12のウォッチドッグパルス入力ポートWDiとを結ぶウォッチドッグドッグパルス線19にウォッチドッグパルス監視回路20を分岐接続し、ウォッチドッグパルスがハイレベル又はロウレベルに固定されたことをもってウォッチドッグパルス監視回路20からシステムの暴走を報知する構成としてある。

【0010】実施例の場合、双方のCPU12, 13

は、相手方のハンドシェーク信号出力ポートPhoからロウレベルのハンドシェーク信号を受けて割り込み要求フラグIFLを立てる。また、データ出力ポートDoは、CPU12, 13内のレジスタ12c, 13cが保持する計数値を供給され、これを相手方のデータ入力ポートDiへと供給する。ウォッチドッグパルス出力ポートWDoは、図2(A), (B)に示したように、1ループが消化される時点でレベルが反転するウォッチドッグパルスを、相手方のウォッチドッグパルス入力ポートWDiに供給する。なお、図2(C), (D)は、異常発生時のウォッチドッグパルスの波形を例示するものである。

【0011】ところで、主CPU12も副CPU13の基本的には、同じ処理プログラムに従って動作する。そこで、以下の説明では、概ね主CPU12に関する処理プログラムに沿って説明する。まず、全体のループ処理プログラムは、図3に示したように、その大部分をステップ(10)で示した演算プログラムが占めており、この演算プログラムに続く判断ステップ(70)において、タイマ12bの計時出力TMがCPU12が保持しているループタイム目標値TLに一致したかどうか判断される。判断の結果、計時出力TMがループタイム目標値TLに一致すると、続くステップ(80)においてウォッチドッグパルスWDmがレベル反転され、相手方の副CPU13へと出力される。さらに、前述のステップ(10)に示した演算プログラムに復帰する前に、ステップ(90)において、自らのウォッチドッグパルスWDmと相手方である副CPU13が出力するウォッチドッグパルスWDsとを同期判定し、クロック発振器12a, 13aのクロック周波数ずれや主副のCPU12, 13によってランダムに異なる割り込み処理に要した時間の違い等により生ずる同期ずれを解消して同期出力するための同期処理を行う。

【0012】ステップ(90)におけるウォッチドッグパルス同期処理においては、まず、図4に示す最初のステップ(100)において、同期処理中の割り込みを禁止する。次に、ステップ(101)において、ループタイム目標値TLに対し初期値を設定する。また、これと同時にロウレベルのハンドシェーク信号をポートPhoから出力する。さらにまた、同期時間の計測に同期カウンタとして用いるレジスタ12cを零リセットする。次に、判断ステップ(102)において、ハンドシェーク信号入力ポートPhiの信号レベルから、相手方である副CPU13の割り込み要求フラグIFLが立っているかどうか、すなわち副CPU13が主CPU12からの同期信号すなわちハンドシェーク信号待ちか否かを判定する。

【0013】判定の結果、割り込み要求フラグIFLが立っていてハンドシェーク信号待ちではないこと、すなわち両CPU12, 13が互いに同期していることが判

明した場合は、ステップ(103)において、割り込み要求フラグIFLをクリアし、続くステップ(104)において、ハンドシェーク信号を停止して同期処理を終える。

【0014】一方、判断ステップ(102)において割り込み要求フラグが立っていないこと、すなわち相手方である副CPU13がハンドシェーク信号を出力していないことが明らかになった場合は、ステップ(120)において、同期カウンタであるレジスタ12cが保持する計数値を1だけ増し、続く判断ステップ(121)において、レジスタ12cが計数値として保持する時間データが規定時間を超過しているかどうか、すなわちCPU12, 13が出力するハンドシェーク信号間に時間差があるかどうかを判定する。なお、この規定時間は、一対のCPU12, 13間で許容できるループタイムずれの最大値を規定するものであり、この値を越えてループタイムに差が生じた場合に暴走が生じたものと判定する閾値そのものを指す。

【0015】ステップ(121)における判定の結果、規定時間超過信号が出力された場合、すなわち予め規定された許容可能な規定時間を超過していることが判れば、続く判断ステップ(122)において、ウォッチドッグパルスPwo, Pwiの極性を相互に比較する。そこで、双方のウォッチドッグパルスPwo, Pwiの極性が一致しなかった場合は、相手方である副CPU13が暴走しているものとし、続く暴走処理ステップ(123)に移行する。この暴走処理ステップでは、主CPU12は自ら演算を停止して全出力を停止し、これと同時にウォッチドッグパルスの出力も停止するため、無限ループへと移行する。ただし、ウォッチドッグパルスPwo, Pwiの極性が一致した場合には、ステップ(103)に移行し、次のループに判定を委ねる。

【0016】このように、CPU12は、それぞれのループ処理プログラム途中の所定ステップにおいてループタイムTLが経過したことを判定し、判定時点でウォッチドッグパルスWDmを相手方である副CPU13に出力し、副CPU13が発するウォッチドッグパルスWDSと自ら発したウォッチドッグパルスWDmとが規定時間内に極性一致しない場合は暴走であると判定し、自らのループ処理を停止する。すなわち、主副一対のCPU12, 13間で相手方のループタイムを相互監視し、少なくとも一方に異常があると判断した場合には、自らループ処理を停止するため、自然消滅的に両CPU12, 13が作動を停止する。従って、主副一対のCPU12, 13が出力するウォッチドッグパルスWDm, WDSの一方、すなわちこの実施例ではWDSの極性変化の有無をウォッチドッグパルス監視回路20により外部監視するだけで、両CPU12, 13の異常を発見することができる。

【0017】一方、ステップ(104)において、ハン

ドシェーク信号出力ポートPhoをハイレベルにセットしてハンドシェーク信号を停止した後は、続くステップ(105)において、タイマ12bの計時出力TM、すなわち上記ステップ(70)～(104)までの同期処理に要した時間を、タイマ12bから16ビットの計数値としてレジスタ12cに取り込む。なお、取り込んだ16ビットの計数値AXは、Aが上位8ビットを表し、Xが下位8ビットを表すが、同期異常が発生していないければ、上位8ビットAは零で下位8ビットXだけがデータとして存在する。次に、タイマ12bを零から動作再開させるため、続くステップ(106)において、タイマコントロールフラグTMCFLをクリアした後、タイマコントロールフラグTMCFLをセットしてタイマ12cを再度動作開始させる。

【0018】タイマ12bの動作再開に続き、まずステップ(107)において、主CPU12は内蔵するレジスタ12cが保持する計数値Xをデータ出力ポートDoにセットし、この計数値Xを副CPU13が取りこめるようとする。また、これに並行して副CPU13の側でも、自らが内蔵するレジスタ13cが保持する計数値Xをデータ出力ポートDoにセットするため、主CPU12は副CPU13が同期処理に要した時間に関する計数値Xを取りこめるようになる。ただし、実施例では、副CPU13を主CPU12に準拠させる構成としてあるため、副CPU13が同期処理に要した時間に関する計数値Xは、使用されないまま終わる。次に、ステップ(108)において同期処理を行う。

【0019】ステップ(108)における同期処理は、ステップ(101)～(104)に非常に似ており、その内容はステップ(101)からレジスタ12cのリセット動作を除外し、かつ判断ステップ(102)の判断否定結果をステップ(120)～(123)を経ずに直接帰還させる内容となっている。このため、主副のCPU12, 13は相手方からハンドシェーク信号が得られるまで、すなわちハンドシェーク信号入力ポートPhiを介して受信されるハンドシェーク信号がロウレベルとなって割り込み要求フラグIFLが起立するのを待ち、最後に相手方に输出していたハンドシェーク信号を停止し、同期処理が行われる。

【0020】このように、暴走が発生していない場合には、ステップ(106)において一対のCPU12, 13がループタイムの計時を新たに同時期に再開する。このため、主副一対のCPU12, 13を各専用のクロック発振器12a, 13aで動作させる2クロック2CPUのシステム構成が原因で、各専用のクロック発振器12a, 13a間に生じたクロック周期のずれが堆積してループタイムずれを招来しそうになつても、規定時間内であれば1ループごとに補正でき、従つてクロック周期のずれが突発的に暴走判定を招来するといった不都合を排除することができる。

【0021】同期処理を終えると、主CPU12（副CPU13）は、ステップ（109）において、相手方の副CPU13（主CPU12）のレジスタ13c（12c）が出力ポートD0から出力する計数値Xを、データ入力ポートD1から取り込んでレジスタ12c（13c）の計数値Aとする。ただし、ここでレジスタ12c、13cに取り込まれた相手方の計数値Xは、主CPU12と副CPU13とで異なる処理を行うための判断ステップ（110）の判断結果を受けて、異なる取り扱いを受ける。判断ステップ（110）は、これまで主CPU12を例に説明してきたフローチャートを、ここから主CPU12と副CPU13とで別扱いとするための判断ステップであり、主CPU12についてはステップ（111）を飛び越えて直接ステップ（112）に移行させ、副CPU13についてはステップ（111）を経てステップ（112）へと導く。すなわち、前述の如くここでは副CPU13側を主CPU12に準拠させる方法によっているため、副CPU13のレジスタ13cが保持する計数値Xは、ステップ（111）において上記の計数値Aすなわち主CPU12側の計数値Xに強制的に合致させられる。こうして、副CPU13側の同期を主CPU12側に同調させる前処理が施され、しかるのちループタイムの補正ステップ（112）へと移行する。

【0022】ステップ（112）では、同期処理に要した時間によってループタイムが延びてしまわないよう、ループタイム目標値TLからステップ（70）～（106）までの同期処理に要した時間（X+Y）を減算し、これを新たなループタイム目標値TLとして保持させる。ただし、Yは、システムに固有のステップ処理時間と含む微調整値であり、例えばタイマ12b内のカウンタクロックの計数値に換算して2や3といった数値が用いられる。こうして、主CPU12と副CPU13は、専用のクロック発振器12a、13aのクロック周波数が周波数ずれを多少生じたり、或いは規定時間の範囲内で割り込み処理に要する時間差が生じたとしても、ステップ（104）までにおいて相互に同期をとられ、またステップ（105）以下において同期に要した時間も補正される。このため、最終的にはループタイムの正確な維持が可能である。同期処理の最後は、ステップ（113）において、それまで禁止していた割り込みを許可し、図3に示した演算プログラム（10）に復帰することで締めくくられる。

【0023】このように、それぞれ固有のクロック信号に従って動作する一対のCPU12、13が、ハンドシェーク線14、15とウォッチドッグパルス線18、19を介して一定時間ごとに同期処理を行い、しかも同期処理に統いて主CPU12は自らの同期処理に要した時間（X+Y）をもって、また副CPU13は同期通信線16を介して主CPU12から送られてくる前記時間

（X+Y）をもって、それぞれループタイム目標値TLを補正するため、1ループごとにループタイムを共通の一定時間に規制し、正確に管理されたループタイムのもとCPU12、13の円滑な処理を保証することができる。特に、一対のCPU12、13のそれぞれの処理時間に個体差が存在したり、或いは各CPU12、13の処理途中に挿入される割り込み処理に費やされる時間が異なる場合でも、暴走以外の原因でウォッチドッグパルス監視回路20が作動してしまうことはなく、的確な暴走監視が可能である。

【0024】なお、上記実施例において、主CPU12を基準に副CPU13の同期を強制するようにしたが、副CPU13を基準に主CPU12に同期を強制するシステム構成も可能であり、その場合はデータ通信線17を介して副CPU13から主CPU12にループタイム補正用時間データ（X+Y）が送り込まれる。

【0025】

【発明の効果】以上説明したように、本発明によれば、それがば同周期の固有のクロック信号に基づき、それぞれに固有の割り込み処理を織り混ぜて互いにほぼ同様のループ処理プログラムに従って動作する一対のCPUが、それぞれのループ処理プログラム途中の所定ステップにおいてループ処理に要するループタイムが所定のループタイム目標値に一致したことを判定し、前記ループタイムの計時を新たに同時に再開するよう相互に同期処理を行うとともに、一方のCPUは自らの同期処理に要した時間をもって、また他方のCPUは前記一方のCPUが同期処理に要した前記時間をもって、それぞれ前記ループタイム目標値を補正し、一対のCPUが常に互いに共通する同一のループタイムをもってループ処理を実行するよう監視するようにしたから、1ループごとにループタイムを共通の一定時間に規制し、正確に管理されたループタイムのもと一対のCPUの円滑な処理を保証することができ、そのため一対のそれぞれの処理時間に個体差が存在したり、或いは各CPUの処理途中に挿入される割り込み処理に費やされる時間が異なる場合でも、暴走以外の原因でウォッチドッグパルス監視回路が作動してしまうことはなく、的確な暴走監視が可能である等の優れた効果を奏する。

【図面の簡単な説明】

【図1】本発明のCPU監視装置の一実施例を示すブロック構成図である。

【図2】図1に示した回路各部の信号波形図である。

【図3】図1に示した主CPUの動作を説明するためのフローチャートである。

【図4】図3に示したウォッチドッグパルス同期処理ステップの具体的な内容を示すフローチャートである。

【図5】従来のCPU監視装置の一例を示すブロック構成図である。

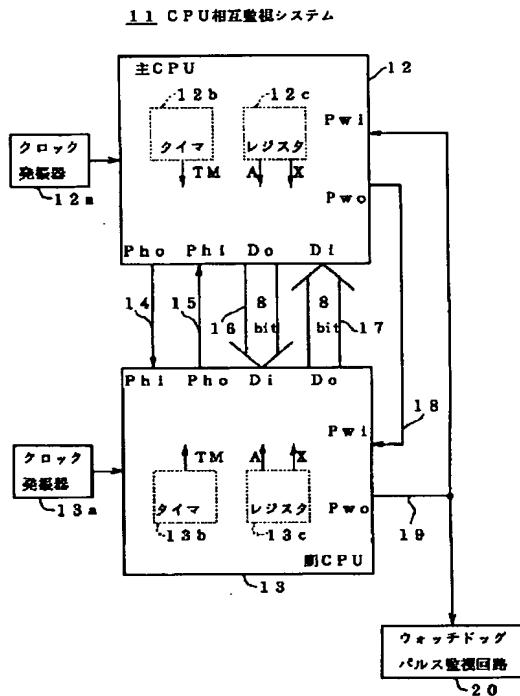
【符号の説明】

9

- 1 1 CPU監視装置
 1 2 主CPU
 1 3 副CPU
 1 2 a, 1 3 a クロック発振器
 1 2 b, 1 3 b タイマ

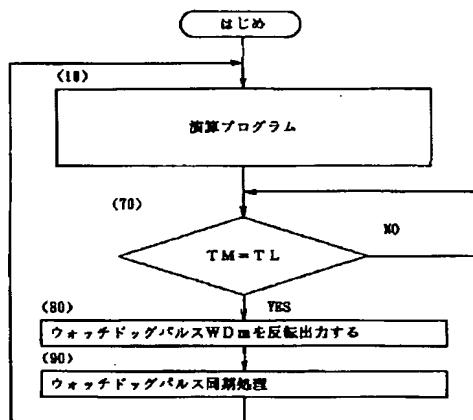
【図1】

— 本発明のCPU監視装置の一実施例を示すブロック構成図



【図3】

— 図1に示した主CPUの動作を説明するためのフローチャート

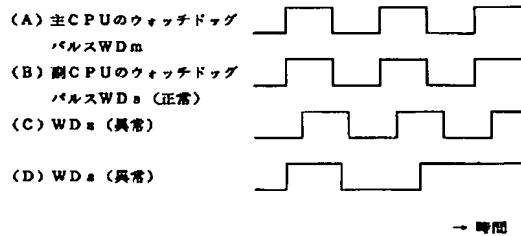


【図2】

- 1 2 c, 1 3 c レジスタ
 1 4, 1 5 ハンドシェーク信号線
 1 6, 1 7 データ通信線
 1 8, 1 9 オッヂドッグパルス線

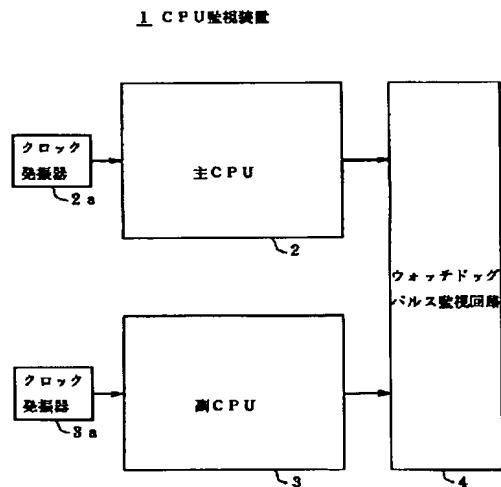
10

— 図1に示した回路各部の信号波形図



【図5】

— 従来のCPU監視装置の一例を示すブロック構成図



【図4】

